# Kerberos TGS Service Ticket Cracking (Kerberoast)

Kerberoast attack was presented 2014 by Tim at Sans HackFest.

This is a easy and stealthy attack that is great to get started with.

## Background and terminology

SPN - Service Principal Name

"A service principal name (SPN) is a unique identifier of a service instance. SPNs are used by Kerberos authentication to associate a service instance with a service logon account. This allows a client application to request that the service authenticate an account even if the client does not have the account name."

SPN is an identifier for a particular service offered by a particular host. The common form for a SPN is the following format: `service class/fqdn@REALM` . So for an example: `IMAP/mail.example.com@EXAMPLE.COM` . So this is the identifier.

Service Principal Name (SPN) is used in the domain to associate the service with a login account. So a SPN is associated with a domain login account. So an account is running the service. The service account runs the service. The service account is usually a Administrator, or belongs to admin group, or the machine. Which is good for an attacker.

- A user wants to interact with a service

1. User sends a request to DC
2. DC responds with TGT
3. User sends TGS request to DC
4. DC sends inter-realm TGT
5. User sends TGS request to DC
6. DC responds with TGS for the specified server. It is encrypted with the servers hash.
7. User sends TGS to server. The server can decrypt the TGS because it is encrypted with a hash that the server knows.

This means that everytime a user tried to access a specific servicer, for example by going to a share, the user recieves a TGS. The user can then offline crack the TGS to find the servers password. An attacker can do this even if the attacker does not have access to the service.

You can see all the TGS-tickets that is on a computer by simply running this command:

```
klist
```

You will not see the actual TGS, as it is stored in memory. So the OLD-SCHOOL attack scenario was:

1. Request access to all services on the domain
2. Extact the TGS from memory
3. Crack the TGS

This was simplified. but some stuff that manages to remove the mimikatz part. No need to actually retrieve the TGS from memory, as it can be retrieved straight from the request.

```
klist
```

Offline brute force of passwords of service accounts with service tickets - No risk of detection - No account lockouts

## How to test

Using impacket:

```
python GetUserSPNs.py -dc-ip 192.168.66.87 -request hackdomain.local/FilipAdmin
```

The resulting hashes are already in hashcat format.

Using invoke-kerberoast from PowerView:

```
Invoke-Kerberoast -OutputFormat Hashcat | Select-Object Hash | Out-File -filepath ‘c:\users\public\HashCapture.txt’ -Width 8000
```

## Crack the hashes

Remember to add some words that are specific to the domain. Maybe company name and stuff like that. Look at the services that are running.

See the chapter on cracking hashes for how to crack the hashes.

## Generate passwordlist

See the chapter on Generate password list for how to generate custom password lists.

### Brute force

If you want to do a brute force attack it is a good idea to first understand the AD password requirements. You can check the minimum pasword length by running this command:

```
 net accounts /domain
net accounts
```

- References https://room362.com/post/2016/kerberoast-pt2/ https://room362.com/post/2016/kerberoast-pt1/